****

**Remote Line Editor For Text Files**

**Software Requirement Specification (SRS) Document**

**Sprint 2 Implementation**

**Project Timeline: 11.10.2022 to 17.10.2022**

**INDEX**

1. Introduction

1.1 Purpose -------------------------------------------------- 4

1.2 Intended audience -------------------------------------------------- 4

1.3 Intended use -------------------------------------------------- 4

1.4 Scope -------------------------------------------------- 4

2. Overall description -------------------------------------------------- 5

2.1 Assumptions and dependency ------------------------------------------------ 5

3. System feature and requirements ------------------------------------------------- 5

3.1 Functionality

* + 1. RLET\_01 : Client Server Connection
    2. RLET\_02 : User Authentication
    3. RLET\_03 : Server Handshaking
    4. RLET\_04 : Server side Firewall Protection
    5. RLET\_05 : Server side Security
    6. RLET\_06 : Service Starting at Boot
    7. RLET\_07 : Anonymous user access:
    8. RLET\_08 : Authenticated User Access
    9. RLET\_9 : Server side Concurrency
    10. RLET\_10 : Client side program starting
    11. RLET\_11 : Client Side Connection request
    12. RLET\_12 : Client Side Authentication
    13. RLET\_13 : Client Side User Environment
    14. RLET\_14 : Client Side Environment Customization
    15. RLET\_15 : Client Side Browsing
    16. RLET\_16 : Client Side File Selection
    17. RLET\_17 : Client Side File Printing
    18. RLET\_18 : Client Side File line Selection
    19. RLET\_19 : Client Side line editing
    20. RLET\_20 : Client Side Quit

3.3 System requirement -------------------------------------------------- 7

3.3.1 Tools to be used -------------------------------------------------- 7

3.4 Non-Functional Requirements ------------------------------------------------- 7

3.5 System feature -------------------------------------------------- 8

4. Data Flow Diagram

4.1 DFD level 0 ------------------------------------------------ 8

4.2 DFD level 1 ------------------------------------------------ 9

**1.** **Introduction: -**

The introduction of the software requirement specification provides an overview of the entire software. The entire SRS with overview description purpose, scope, tools used and basic description. The aim of this document is to gather, analyze and give an in-depth insight into the complete Remote Line Editor For Text Files application by defining the problem statement in detail. The detailed requirements of the Remote Line Editor For Text Files application is provided in this document.

**1.1** **Purpose**: **-**The purpose of this document is to show the requirements for the Remote Line Editor For Text Files application, which can be useful to edit text files by the user.

**1.2** **Intended Audience: -**This application is intended to be use by, User.

**1.3** **Intended Use: -**

* Development Team
* Maintenance Team
* Clients

Since this a general-Purpose Software any one can access it.

**1.4** **Scope: -** This application can be very helpful to user for editing his text files from any device having internet. This application does not require any installation. So, with this application user can edit his text files at less time with his credentials(User Id & Password)

**2. Overall Description: -**

It is Remote Line Editor application used to edit text files by user with his credentials(User Id & Password). With this application multiple users can edit the text files from any device having internet connection with out any installation. This best thing about this application is it help users to produce an error free result within a few seconds. This application is compatible with almost every smart device having an internet connection. This application don’t require any installation process and provides the best options for editing tasks.

**2.1 Assumptions and Dependency: -**

* This application is compatible with almost every smart device having an internet connection.

**3.System Features and Requirements: -**

**3.1 Functionality: -**

* + 1. **RLET\_01| Client Server Connection:** A connection between client and server must be established first.
    2. **RLET\_02| User Authentication:** The server should authenticate the user from a poll of registered user data kept with appropriate data structure.
    3. **RLET\_03| Server Handshaking:** The user should get the customize message from server entering the user id but before providing the password by which the user get conformation that it got connected to the proper server.
    4. **RLET\_04| Server side Firewall Protection:** The server should listen to port 8012 for client connection and make the client side computer access the server port.
    5. **RLET\_05| Server side Security:** Before allowing a client connection the server should match the client IP with the blacklist IP specified in /etc/phone/ client\_blacklist .
    6. **RLET\_06| Service Starting at Boot:** The server must configure to start the service automatically at boot.
    7. **RLET\_07| Anonymous user access:** Server should support a virtual user with name anonymous to add contact to public group only without any password.
    8. **RLET\_08| Authenticated User Access:** All the authenticated user should be allowed to edit any files from their home directory.
    9. **RLET\_09| Server side Concurrency:** The server should support the concurrent client connection.
    10. **RLET\_10| Client side program starting:** The user should start the client program to connect the Server whenever he wants.
    11. **RLET\_11| Client Side Connection request:** Client should request a connection to server on port 8012.
    12. **RLET\_12| Client Side Authentication:** Client program should start with an authentication (default to anonymous).
    13. **RLET\_13| Client Side User Environment:** On successful authentication the user should place on his home directory as PWD & anonymous user PWD should be/var/edit.
    14. **RLET\_14| Client Side Environment Customization:** User can use cd to change his remote PWD on client side.
    15. **RLET\_15| Client Side Browsing:** User can use ls to browse the file list in the remote PWD on client side.
    16. **RLET\_16| Client Side File Selection:** User must use select FILENAME sub command to select a file for editing.
    17. **RLET\_17| Client Side File Printing:** User must use print [START][END] sub command to print a part or the whole file with the line number.
    18. **RLET\_18| Client Side line Selection:** user must use the edit LINENUM sub command to get the content of the line to edit.
    19. **RLET\_19| Client Side line editing:** User must press enter after changes to accept the editing and save it to the file.
    20. **RLET\_20| Client Side Quit:** The user should use a sub command bye to terminate the connection.

**3.2 Technical Requirements: -**

### **3.2.1. Tools to be used:**

* Unistd.h library for fork
* C File Handling
* C Language

**3.3 Non-Functional Requirements:**

* C Unit to automate unit testing
* Unit Testing and Integrated Testing Report
* Valgrind to detect memory leak
* Make file
* Multi file compilation and execution process using makefile.
* Gcoverage
* Gprof ( Flat Profile and Call graph )

**3.4 System Features: -**

### Supportability:The system is easy to use.

* Time Saving: The best thing about these online editors is that they help users to produce an error-free result with in a few seconds.
* Easily Accessible: This application is compatible with almost every smart device having an internet connection.
* Reliability & Availability**:** The system is available 24/7 that is whenever the user would like to use the system, they can use it up to its functionalities.

### Performance: The system will work on the user’s terminal**.**

### Portability: Portability to multiple operating systems.

**4. DataFlow Diagram:**

**4.DFDLevel0**A picture containing table

Description automatically generated

**4.1 DFD Level 1 –**

Diagram

Description automatically generated